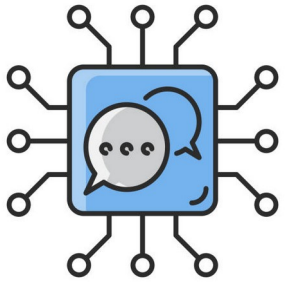
Shallow Parsing and Tools for NLP

Tushar B. Kute,
http://tusharkute.com



Morphological Analysis

- Morphological analysis is the process of examining possible resolutions to unquantifiable, complex problems involving many factors.
- The root of the word morphology comes from the Greek word, morphe, for form.
- Morphological analysis takes a problem with many known solutions and breaks them down into their most basic elements, or forms, in order to more completely understand them.

Morphological Analysis

- Morphological analysis is used in general problem solving, linguistics and biology.
- In many fields of study morphology facilitates clearer instruction for teachers to help students understand problems and their solutions.
- For general problem solving, morphological analysis provides a formalized structure to help examine the problem and possible solutions.
- The elements of a problem and its solutions are arranged in a matrix to help eliminate illogical solutions.

Morphological Analysis

- In linguistics, words are broken down into the smallest units of meaning: morphemes.
- Morphemes can sometimes be words themselves as in the case of free morphemes, which can stand on their own.
- Other morphemes can add meaning but not stand as words on their own; bound morphemes need to be used along with another morpheme to make a word.
- Cats, for example, is a two-morpheme word. Its base, cat, is a free morpheme and its suffix an s, to denote pluralization, a bound morpheme.

Morphological Analysis

- As a school of thought morphology is the creation of astrophysicist Fritz Zwicky. Zwicky contrived the methodology to address non quantified problems that have many apparent solutions.
- For problems to be suited to morphological analysis they are generally inexpressible in numbers.
- Other problems are better addressed with the more traditional decomposition method where complexity is broken down in parts and trivial elements are ignored to produce a simplified problem and solution.

Morphological Analysis

- Practical

Tokenization

- The first thing you need to do in any NLP project is text preprocessing.
- Preprocessing input text simply means putting the data into a predictable and analyzable form. It's a crucial step for building an amazing NLP application.
- There are different ways to preprocess text:
  - stop word removal,
  - tokenization,
  - stemming.

Tokenization

- Tokenization is the first step in any NLP pipeline. It has an important effect on the rest of your pipeline.
- A tokenizer breaks unstructured data and natural language text into chunks of information that can be considered as discrete elements.
- The token occurrences in a document can be used directly as a vector representing that document.
- This immediately turns an unstructured string (text document) into a numerical data structure suitable for machine learning.

Tokenization

Types of Tokenizers

- Word Tokenizer
- Sentence Tokenizer
- White Space Tokenizer
- Word Tokenizer
- Space Tokenizer
- Tab Tokenizer
- Line Tokenizer
- Tree Bank Word Tokenizer
- Tweet Tokenizer
- MWET tokenizer

Types of Tokenizers

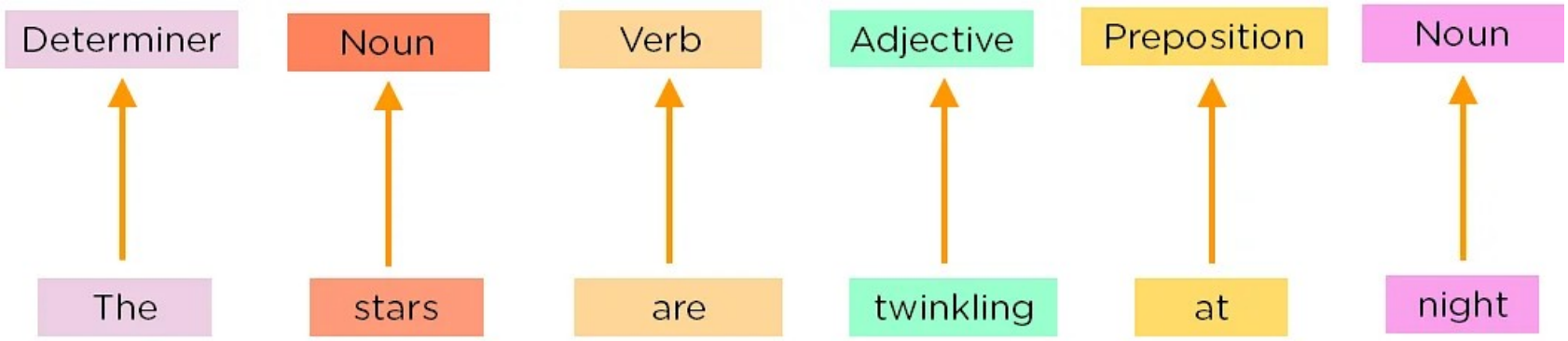
- Practical

POS Tagging

- Part-of-speech (POS) tagging is the process of assigning a word to its grammatical category, in order to understand its role within the sentence. Traditional parts of speech are nouns, verbs, adverbs, conjunctions, etc.
- Part-of-speech taggers typically take a sequence of words (i.e. a sentence) as input, and provide a list of tuples as output, where each word is associated with the related tag.
- Part-of-speech tagging is what provides the contextual information that a lemmatiser needs to choose the appropriate lemma.

Part of Speech Tagging

- Now, you must explain the concept of nouns, verbs, articles, and other parts of speech to the machine by adding these tags to our words. This is called ‘part of’.



POS Tags

- CC coordinating conjunction
- CD cardinal digit
- DT determiner
- EX existential there (like: “there is” … think of it like “there exists”)
- FW foreign word
- IN preposition/subordinating conjunction
- JJ adjective ‘big’
- JJR adjective, comparative ‘bigger’
- JJS adjective, superlative ‘biggest’
- LS list marker 1)
- MD modal could, will
- NN noun, singular ‘desk’
- NNS noun plural ‘desks’
- NNP proper noun, singular ‘Harrison’
- NNPS proper noun, plural ‘Americans’
- PDT predeterminer ‘all the kids’
- POS possessive ending parent’s
- PRP personal pronoun I, he, she
- PRP$ possessive pronoun my, his, hers
- RB adverb very, silently,
- RBR adverb, comparative better
- RBS adverb, superlative best
- RP particle give up
- TO, to go ‘to’ the store.
- UH interjection, errrrrrrrrm
- VB verb, base form take
- VBD verb, past tense took
- VBG verb, gerund/present participle taking
- VBN verb, past participle taken
- VBP verb, sing. present, non-3d take
- VBZ verb, 3rd person sing. present takes
- WDT wh-determiner which
- WP wh-pronoun who, what
- WP$ possessive wh-pronoun whose
- WRB wh-abverb where, when

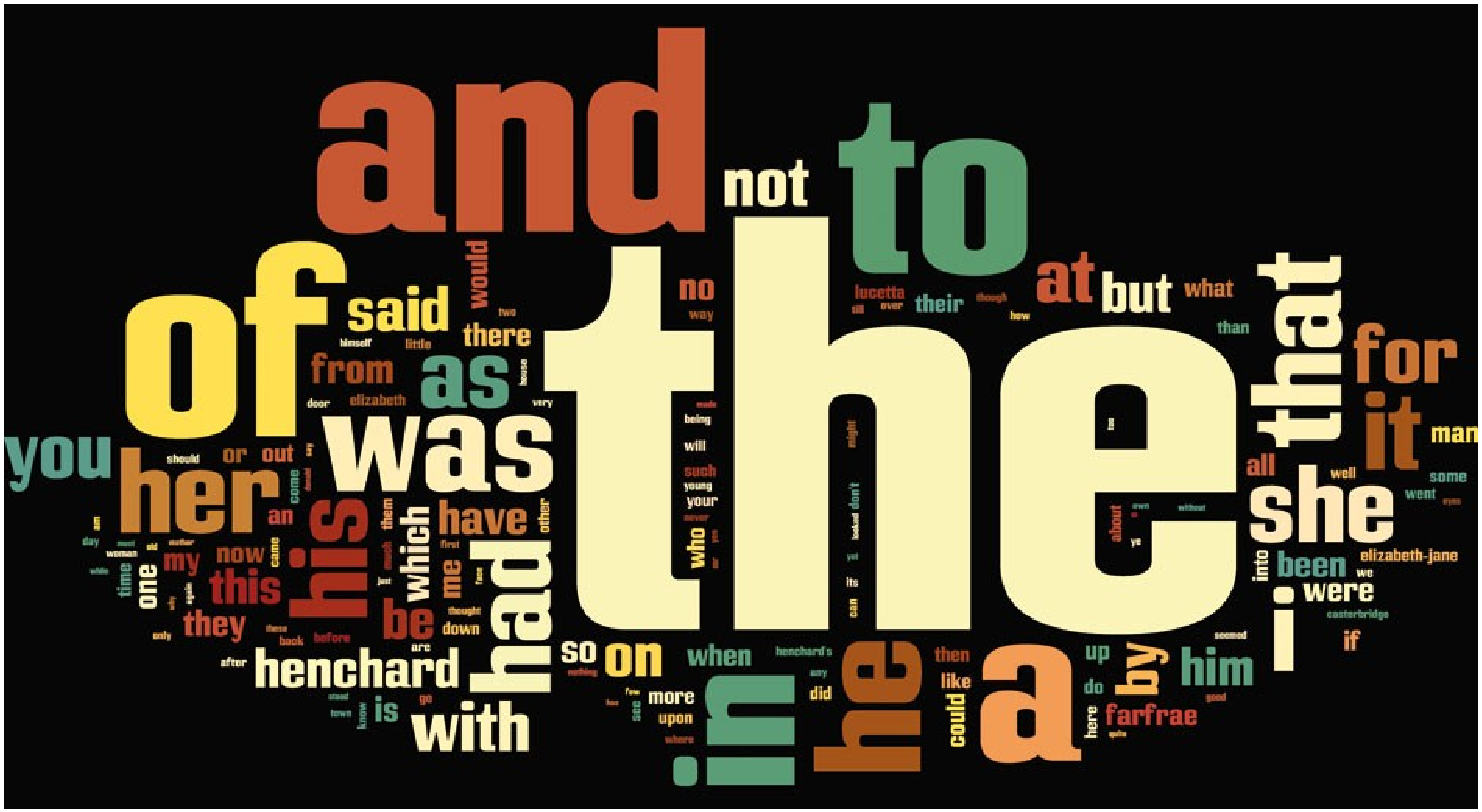
POS Tags

- Practical

Stopwords

- The words which are generally filtered out before processing a natural language are called stop words.
- These are actually the most common words in any language (like articles, prepositions, pronouns, conjunctions, etc) and does not add much information to the text.
- Examples of a few stop words in English are “the”, “a”, “an”, “so”, “what”.

Stopwords



Why to remove stopwords?

- Stop words are available in abundance in any human language.
- By removing these words, we remove the low-level information from our text in order to give more focus to the important information.
- In order words, we can say that the removal of such words does not show any negative consequences on the model we train for our task.
- Removal of stop words definitely reduces the dataset size and thus reduces the training time due to the fewer number of tokens involved in the training.

Do we remove stopwords always?

- The answer is no!
- We do not always remove the stop words. The removal of stop words is highly dependent on the task we are performing and the goal we want to achieve.
- For example, if we are training a model that can perform the sentiment analysis task, we might not remove the stop words.
- Movie review: “The movie was not good at all.”
- Text after removal of stop words: “movie good”

Stopwords Removal

- Practical

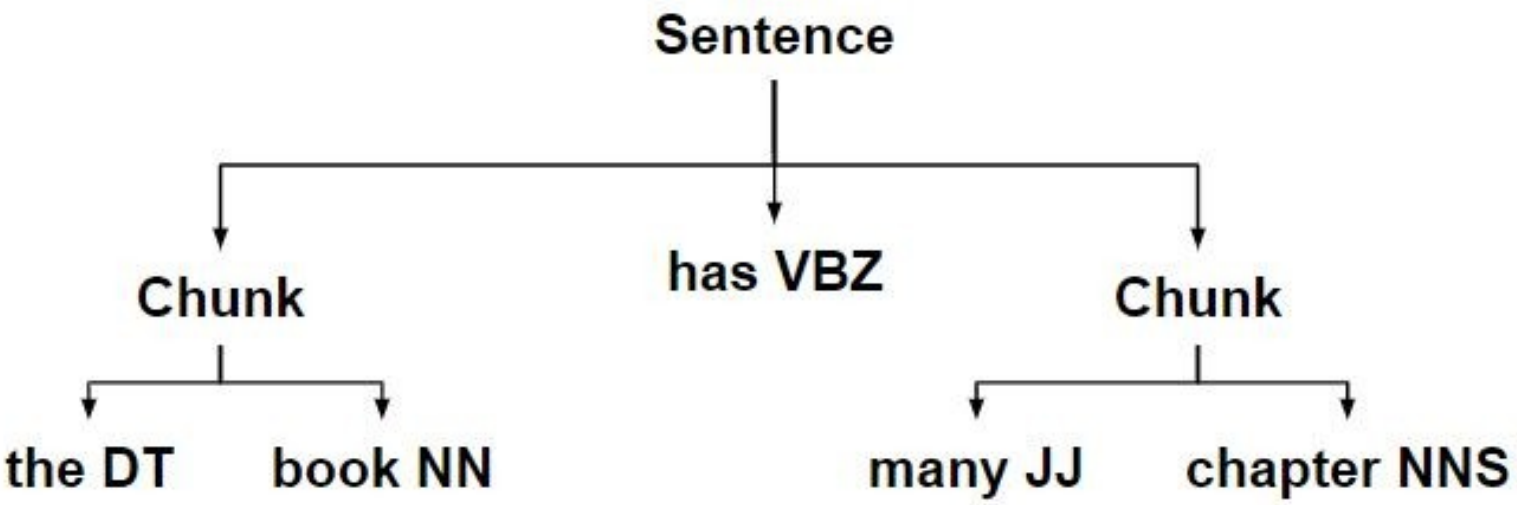
Chunking

- Chunking is defined as the process of natural language processing used to identify parts of speech and short phrases present in a given sentence.
- Recalling our good old English grammar classes back in school, note that there are eight parts of speech namely the noun, verb, adjective, adverb, preposition, conjunction, pronoun, and interjection.
- Also, in the above definition of chunking, short phrases refer to the phrases formed by including any of these parts of speech.

Chunking

- For example, chunking can be done to identify and thus group noun phrases or nouns alone, adjectives or adjective phrases, and so on. Consider the sentence below:
  - “I had burgers and pastries for breakfast.”
- In this case, if we wish to group or chunk noun phrases, we will get “burgers”, “pastries” and “lunch” which are the nouns or noun groups of the sentence.

Chunking


Sentence
Chunk
has VBZ
Chunk
the DT
book NN
many JJ
chapter NNS

Chunking: Where it is used?

- Why would we want to learn something without knowing where it is widely used?!
- Chunking is used to get the required phrases from a given sentence.
- However, POS tagging can be used only to spot the parts of speech that every word of the sentence belongs to.
- When we have loads of descriptions or modifications around a particular word or the phrase of our interest, we use chunking to grab the required phrase alone, ignoring the rest around it.

Chunking: Types

- Chunking up:
  - Here, we don’t dive deep; instead, we are happy with just an overview of the information. It just helps us get a brief idea of the given data.
- Chunking down:
  - Unlike the previous type of chunking, chunking down helps us get detailed information.
- So, if you just want an insight, consider “chunking up” otherwise prefer “chunking down”.

Chunking

- Practical

Named Entity Recognition

- Named entity recognition (NER) — sometimes referred to as entity chunking, extraction, or identification — is the task of identifying and categorizing key information (entities) in text.
- An entity can be any word or series of words that consistently refers to the same thing. Every detected entity is classified into a predetermined category.
- For example, an NER machine learning (ML) model might detect the word “MITU Skillologies” in a text and classify it as a “Company”.

Named Entity Recognition

- NER is a form of natural language processing (NLP), a subfield of artificial intelligence.
- NLP is concerned with computers processing and analyzing natural language, i.e., any language that has developed naturally, rather than artificially, such as with computer coding languages.

Named Entity Recognition

contentSkip to site indexPoliticsSubscribeLog InSubscribeLog InToday's PaperAdvertisementSupported ORG byF.B.I. Agent Peter Strzok PERSON , Who Criticized Trump PERSON in Texts, Is FiredImagePeter Strzok, a top F.B.I. GPE counterintelligence agent who was taken off the special counsel investigation after his disparaging texts about President Trump PERSON were uncovered, was fired. CreditT.J. Kirkpatrick PERSON for The New York TimesBy Adam Goldman ORG and Michael S. SchmidtAug PERSON . 13 CARDINAL , 2018WASHINGTON CARDINAL — Peter Strzok PERSON , the F.B.I. GPE senior counterintelligence agent who disparaged President Trump PERSON in inflammatory text messages and helped oversee the Hillary Clinton PERSON email and Russia GPE investigations, has been fired for violating bureau policies, Mr. Strzok PERSON 's lawyer said Monday DATE .Mr. Trump and his allies seized on the texts — exchanged during the 2016 DATE campaign with a former F.B.I. GPE lawyer, Lisa Page — in PERSON assailing the Russia GPE investigation as an illegitimate "witch hunt." Mr. Strzok PERSON , who rose over 20 years DATE at the F.B.I. GPE to become one of its most experienced counterintelligence agents, was a key figure in the early months DATE of the inquiry.Along with writing the texts, Mr. Strzok PERSON was accused of sending a highly sensitive search warrant to his personal email account.The F.B.I. GPE had been under immense political pressure by Mr. Trump PERSON to dismiss Mr. Strzok PERSON , who was removed last summer DATE from the staff of the special counsel, Robert S. Mueller III PERSON . The president has repeatedly denounced Mr. Strzok PERSON in posts on

Named Entity Recognition

- Person
  - E.g., Elvis Presley, Audrey Hepburn, David Beckham
- Organization
  - E.g., Google, Mastercard, University of Oxford
- Time
  - E.g., 2006, 16:34, 2am
- Location
  - E.g., Trafalgar Square, MoMA, Machu Picchu
- Work of art
  - E.g., Hamlet, Guernica, Exile on Main St.

How NER used?

- NER is suited to any situation in which a high-level overview of a large quantity of text is helpful.
- With NER, you can, at a glance, understand the subject or theme of a body of text and quickly group texts based on their relevancy or similarity.

How NER used?

- Human resources
  - Speed up the hiring process by summarizing applicants' CVs; improve internal workflows by categorizing employee complaints and questions
- Customer support
  - Improve response times by categorizing user requests, complaints and questions and filtering by priority keywords

How NER used?

- Search and recommendation engines
  - Improve the speed and relevance of search results and recommendations by summarizing descriptive text, reviews, and discussions
  - Booking.com is a notable success story here
- Content classification
  - Surface content more easily and gain insights into trends by identifying the subjects and themes of blog posts and news articles

How NER used?

- Health care
  - Improve patient care standards and reduce workloads by extracting essential information from lab reports
  - Roche is doing this with pathology and radiology reports
- Academia
  - Enable students and researchers to find relevant material faster by summarizing papers and archive material and highlighting key terms, topics, and themes
  - The EU's digital platform for cultural heritage, Europeana, is using NER to make historical newspapers searchable

NER

- Practical

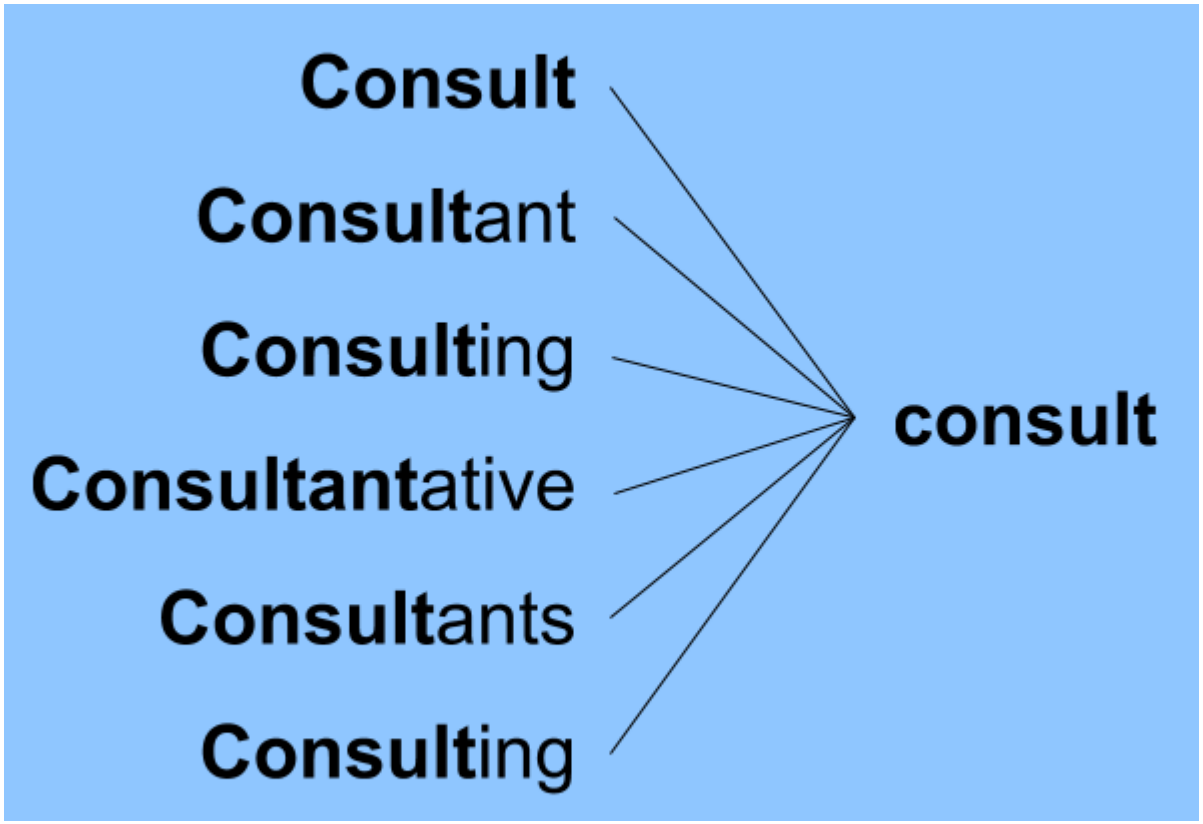
Stemming

- Stemming is the process of reducing a word to its stem that affixes to suffixes and prefixes or to the roots of words known as "lemmas".
- Stemming is important in natural language understanding (NLU) and natural language processing (NLP).
- Stemming is a part of linguistic studies in morphology as well as artificial intelligence (AI) information retrieval and extraction.

Stemming

- Stemming and AI knowledge extract meaningful information from vast sources like big data or the internet since additional forms of a word related to a subject may need to be searched to get the best results.
- Stemming is also a part of queries and internet search engines.
- Recognizing, searching and retrieving more forms of words returns more results.
- When a form of a word is recognized, it's possible to return search results that otherwise might have been missed.

Stemming


Consult
Consultant
Consulting
Consultantative
Consultants
Consulting
consult

Stemming

- Practical

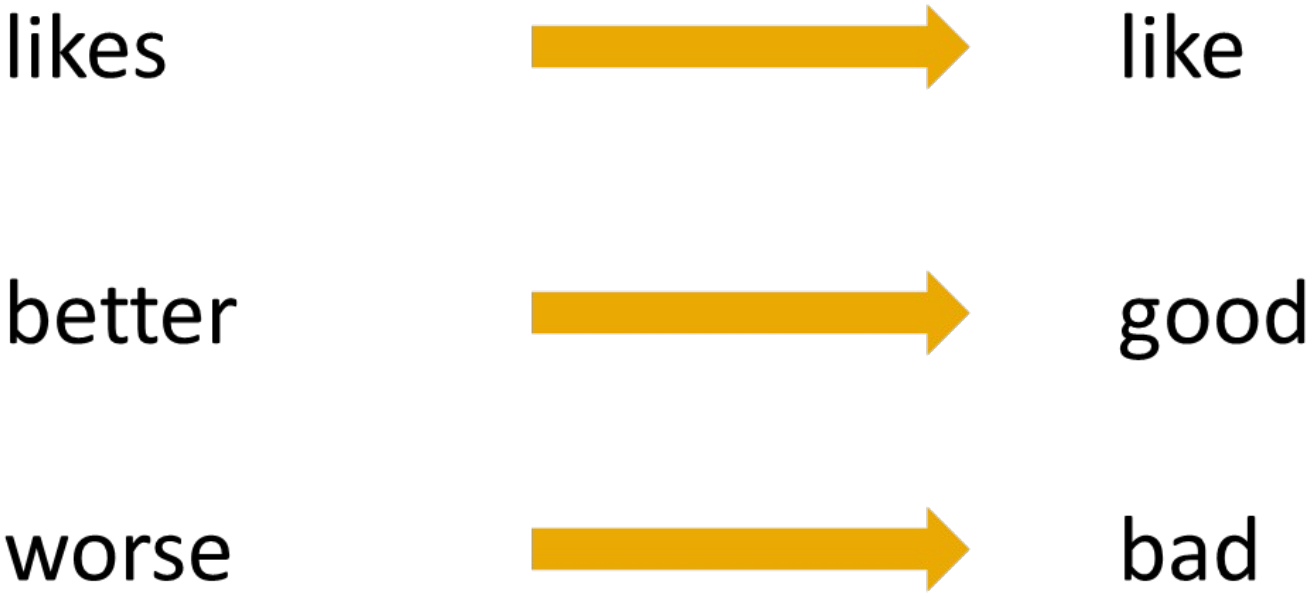
Stemming

Word	Porter	Lancaster	Lemmatiser
wrote	wrote	wrot	write
thinking	think	think	think
remembered	rememb	rememb	remember
relies	reli	rely	rely
ate	ate	at	eat
gone	gone	gon	go
won	won	won	win
ran	ran	ran	run
swimming	swim	swim	swim
mistreated	mistreat	mist	mistreat

Lemmatization

- Lemmatization is a text normalization technique used in Natural Language Processing (NLP), that switches any kind of a word to its base root mode.
- Lemmatization is responsible for grouping different inflected forms of words into the root form, having the same meaning.

Lemmatization



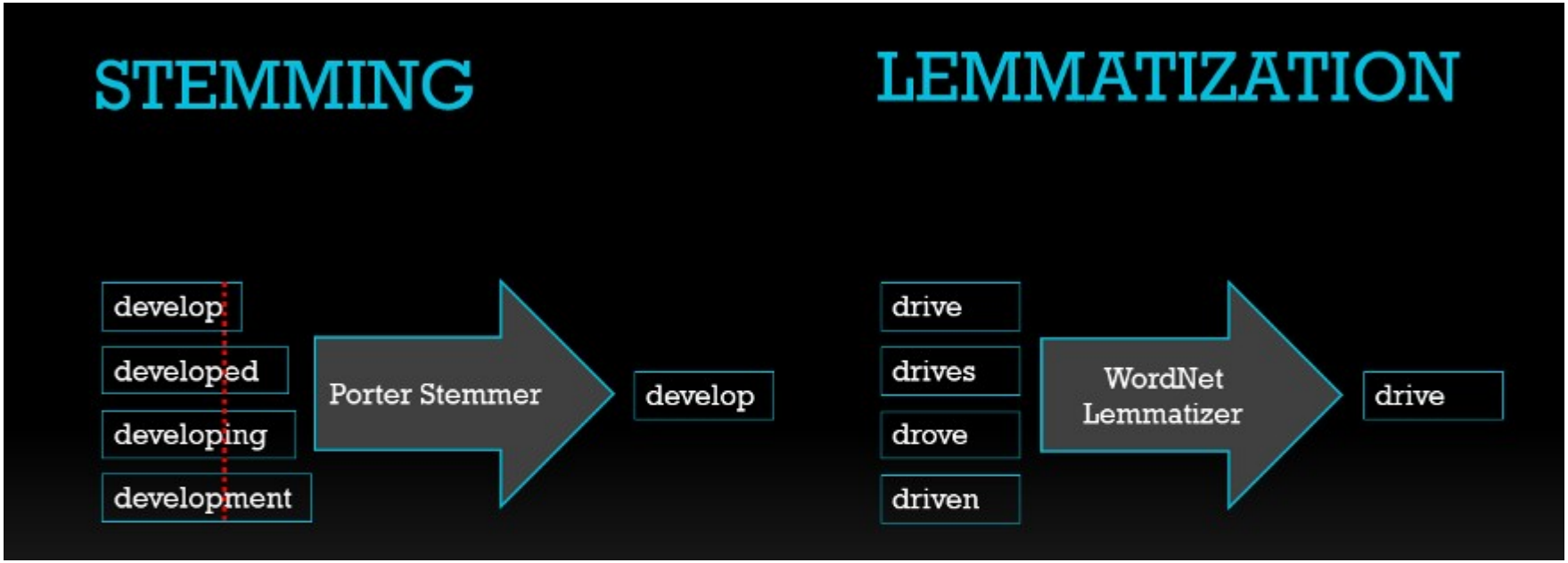
Lemmatization

- Lemmatization is among the best ways to help chatbots understand your customers' queries to a better extent.
- Since this involves a morphological analysis of the words, the chatbot can understand the contextual form of the words in the text and can gain a better understanding of the overall meaning of the sentence that is being lemmatized.
- Lemmatization is also used to enable robots to speak and converse. This makes lemmatization a rather important part of natural language understanding (NLU) and natural language processing (NLP) in artificial intelligence.

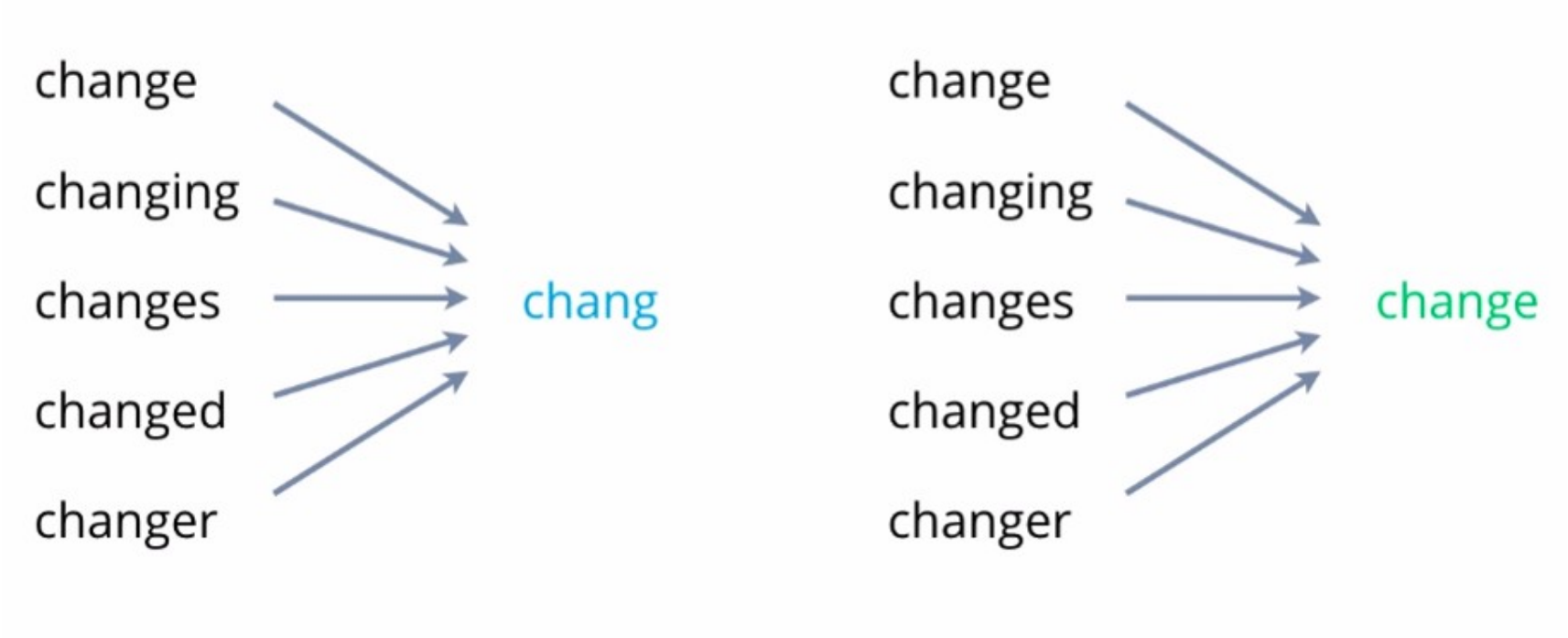
Lemmatization

- Lemmatization is a vital part of Natural Language Understanding (NLU) and Natural Language Processing (NLP). It plays critical roles both in Artificial Intelligence (AI) and big data analytics.
- Lemmatization is extremely important because it is far more accurate than stemming.
- This brings great value when working with a chatbot where it is crucial to understand the meaning of a user's messages.
- The major disadvantage to lemmatization algorithms, however, is that they are much slower than stemming algorithms.

Stemming vs. Lemmatization



Stemming vs. Lemmatization



Word Sense Disambiguation

- Word Sense Disambiguation is an important method of NLP by which the meaning of a word is determined, which is used in a particular context.
- NLP systems often face the challenge of properly identifying words, and determining the specific usage of a word in a particular sentence has many applications.
- Word Sense Disambiguation basically solves the ambiguity that arises in determining the meaning of the same word used in different situations.

Word Sense Disambiguation

- WSD can be used alongside Lexicography. Much of the modern Lexicography is corpus-based.
- WSD, used in Lexicography can provide significant textual indicators.
- WSD can also be used in Text Mining and Information Extraction tasks.
- As the major purpose of WSD is to accurately understand the meaning of a word in particular usage or sentence, it can be used for the correct labeling of words.

Word Sense Disambiguation

- For example, from a security point of view, a text system should be able to understand the difference between a coal “mine” and a land “mine”.
- While the former serves industrial purposes, the latter is a security threat. So a text mining application must be able to determine the difference between the two.
- Similarly, WSD can be used for Information Retrieval purposes. Information Retrieval systems work through text data primarily based on textual information. Knowing the relevance of using a word in any sentence will surely help.

Lesk Algorithm

- Lesk Algorithm is a classical Word Sense Disambiguation algorithm introduced by Michael E. Lesk in 1986.
- The Lesk algorithm is based on the idea that words in a given region of the text will have a similar meaning.
- In the Simplified Lesk Algorithm, the correct meaning of each word context is found by getting the sense which overlaps the most among the given context and its dictionary meaning.

Lesk Algorithm

- Practical

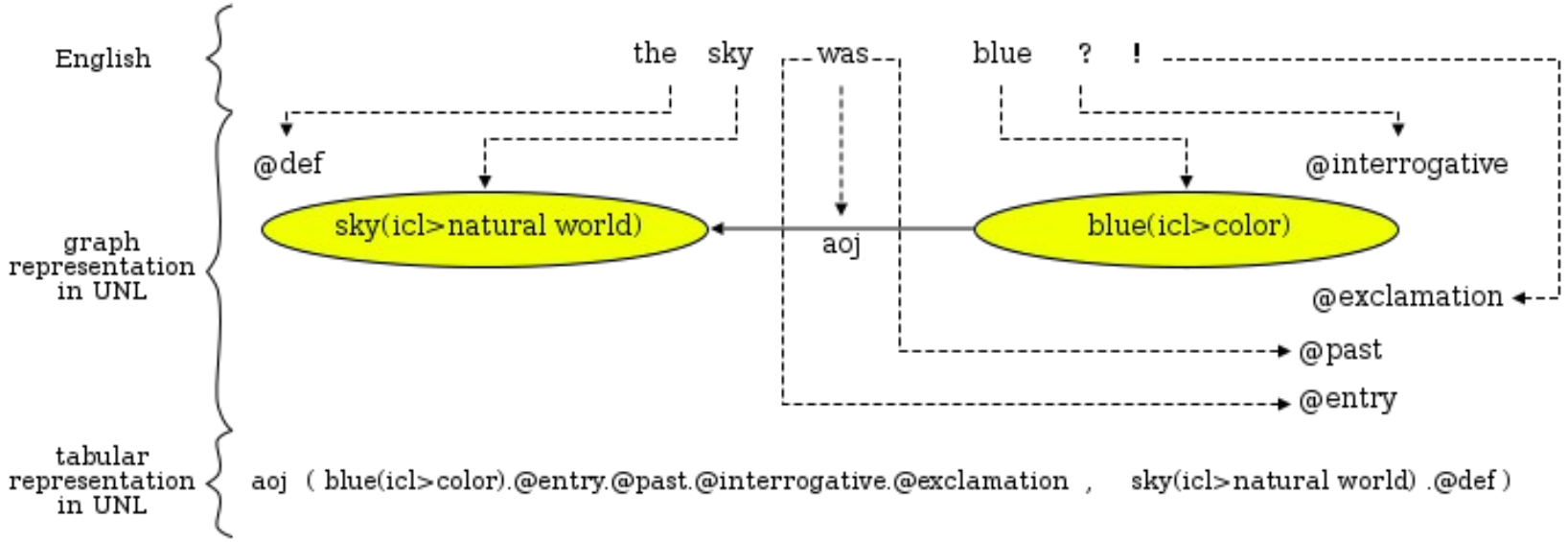
Universal Networking Language

- Universal Networking Language (UNL) is a declarative formal language specifically designed to represent semantic data extracted from natural language texts.
- It can be used as a pivot language in interlingual machine translation systems or as a knowledge representation language in information retrieval applications.

Universal Networking Language

- UNL is designed to establish a simple foundation for representing the most central aspects of information and meaning in a machine- and human-language-independent form.
- As a language-independent formalism, UNL aims to code, store, disseminate and retrieve information independently of the original language in which it was expressed.
- In this sense, UNL seeks to provide tools for overcoming the language barrier in a systematic way.

Universal Networking Language



UNL India

- https://www.cfilt.iitb.ac.in

Synonyms and Antonyms

- NLTK Wordnet can be used to find synonyms and antonyms of words.
- NLTK Corpus package is used to read the corpus to understand the lexical semantics of the words within the document.
- A WordNet involves semantic relations of words and their meanings within a lexical database.
- The semantic relations within the WordNet are hypernyms, synonyms, holonyms, hyponyms, meronyms.
- NLTK WordNet includes the usage of synsets for finding the words within the WordNet with their usages, definitions, and examples.

Synonyms

- To find the synonyms of a word with NLTK WordNet, the instructions below should be followed.
  - Import NLTK.corpus
  - Import WordNet from NLTK.Corpus
  - Create a list for assigning the synonym values of the word.
  - Use the “synsets” method.
  - use the “syn.lemmas” property to assign the synonyms to the list with a for loop.
  - Call the synonyms of the word with NLTK WordNet within a set.

Antonyms

- To find the Antonyms of a Word with NLTK WordNet and Python, the following instructions should be followed.
  - Import NLTK.corpus
  - Import WordNet from NLTK.Corpus
  - Create a list for assigning the synonym values of the word.
  - Use the “synsets” method.
  - use the “syn.lemmas” property to assign the synonyms to the list with a for loop.
  - Use the “antonyms()” method with “name” property for calling the antonym of the phrase.
  - Call the antonyms of the word with NLTK WordNet within a set.

POS Tagging for Synonym and Antonym

- To find the Antonyms of a Word with NLTK WordNet and Python, the following instructions should be followed.
  - Import NLTK.corpus
  - Import WordNet from NLTK.Corpus
  - Create a list for assigning the synonym values of the word.
  - Use the “synsets” method.
  - use the “syn.lemmas” property to assign the synonyms to the list with a for loop.
  - Use the “antonyms()” method with “name” property for calling the antonym of the phrase.
  - Call the antonyms of the word with NLTK WordNet within a set.

POS Tagging for Indian Languages

```
# -*- coding: utf-8 -*-
from nltk.corpus import indian
from nltk.tag import tnt
import nltk

print 'Indian File IDs: ', indian.fileids()

print 'Number of Characters:'
for ch in indian.fileids():
    print ch
    print len(indian.raw(ch))
```

POS Tagging for Indian Languages

```
print 'Number of Words:'
for wd in indian.fileids():
    print wd
    print len(indian.words(wd))

print 'Number of Sentences:'
for st in indian.fileids():
    print st
    print len(indian.sents(st))

sents = indian.sents('marathi.pos')
for sen in sents:
    print sen[0]
```

POS Tagging in Marathi

```
pos = indian.tagged_sents('marathi.pos')
for sent in pos:
    print sent[0][0], sent[0][1]

train_data = indian.tagged_sents('marathi.pos')
tnt_pos_tagger = tnt.TnT()
tnt_pos_tagger.train(train_data)

word = 'आणि शिक्षण तत्पूर्वी सुरू केले'
tags = tnt_pos_tagger.tag(nltk.word_tokenize
        (word.decode('utf-8')))
print tags
for tag in tags:
    print 'Word is:', tag[0],'and POS is:',tag[1]
```

POS Tags output

```
Word is: अगि and POS is: CC
Word is: शिक्षण and POS is: NN
Word is: तत्पूर्वी and POS is: PRP
Word is: सुरू and POS is: JJ
Word is: केले and POS is: VM
```

Thank you

@mitu_skillologies

@mITuSkillologies

@mitu_group

@mitu-skillologies

@MITUSkillologies

kaggle

@mituskillologies

Web Resources
https://mitu.co.in
http://tusharkute.com

@mituskillologies

contact@mitu.co.in

tushar@tusharkute.com